RFC: Split Object Layer and View from hdf-java into hdf-view Repository

Allen Byrne

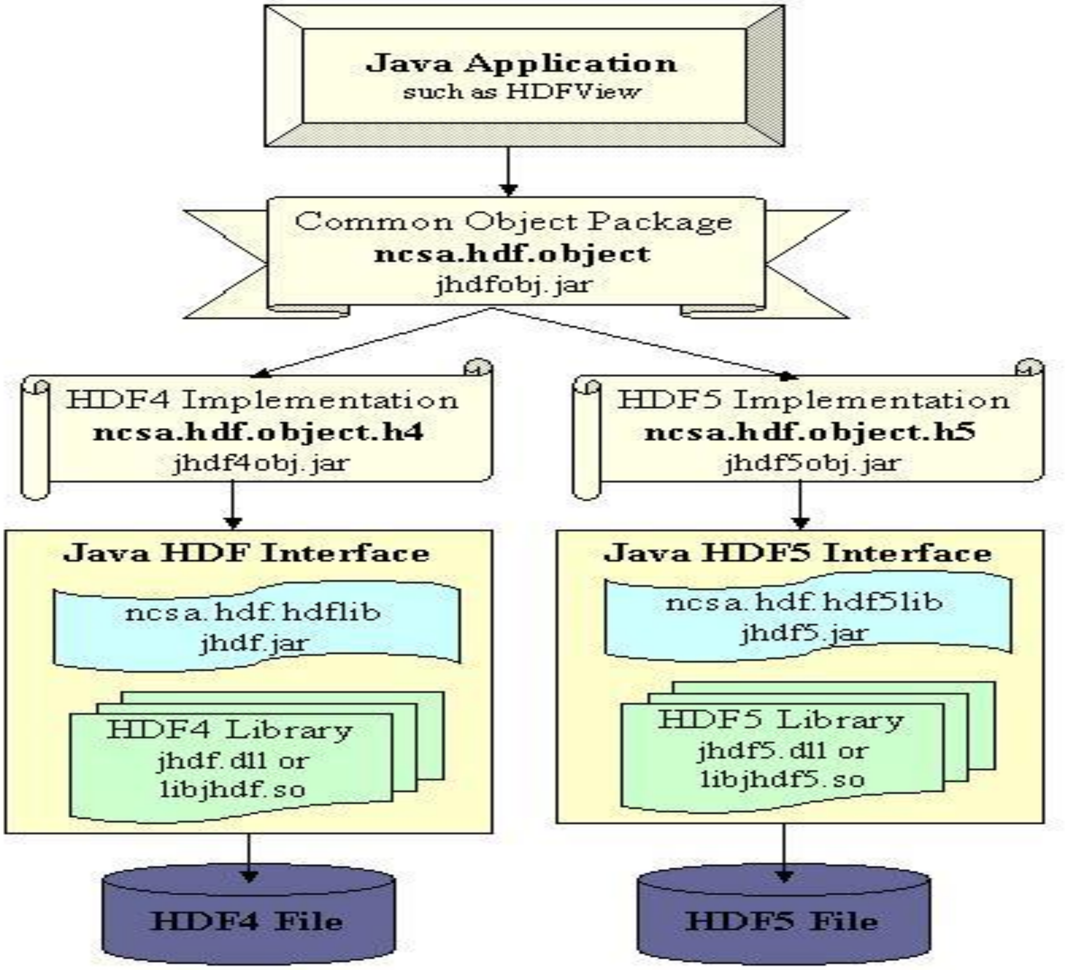
The hdf-java project consists of two major layers; the JNI or native code libraries and the Java class libraries.

The native code layer includes the platform specific interface libraries which wrap the HDF library APIs and the Java jar file which communicates with those interface libraries. This layer uses C for the platform libraries and Java for the interface jar file. This layer is the same for HDF4 as for HDF5.

The Java class libraries are the Java object model (ncsa.hdf.object.*) jar files and the HDFView application. This layer uses only Java for all the class jar files and therefore is platform independent.

I propose to split the Java class libraries, which includes the object model jar files and the HDFView application, into a separate project repository. This project will be named hdf-view. The native code project repository will continue to be named hdf-java

1 Introduction



The hdf-java project consists of two major layers; the JNI or native code libraries and the Java class libraries, each with its own build and test requirements. The native code requires a platform specific C compiler whose output is a library that is used by the Java interface library (or jar). The platform independent Java class libraries use this Java interface library to access the HDF APIs.

The use of CMake has allowed the entire hdf-java project to be built on a variety of platforms using a consistent configuration. However, most users of Java expect to build Java programs using Ant or Maven in a platform independent method.

I propose to split the Java class libraries, which includes the object model jar files and the HDFView application, into a separate project repository. This hdf-view project would not use cmake or autotools, but rely on the Ant build system to create platform independent object libraries and HDFView. The native code libraries will continue to use CMake.

2 Approach

For the next release, 2.11, CMake will continue to be used as the build system. This will allow the release process to create the deliverables required with minimum risk. The intention is that after the release, the native code layer will use CMake while the Java class libraries layer and HDFView will use ant/maven (no CMake).

The ideal build process would have the native libraries exposed as an interface choice within the hdf4 or hdf5 products. This will be proposed at a future time. For the upcoming release, the native libraries can continue to be built as a separate product in sync with the hdf4/hdf5 releases. The hdf-view project releases can be incorporated into the Java Maven community and released on separate schedule.

Revision History

August 4, 2014:	Version 1 circulated for comment within The HDF Group.
August 12, 2014	Add image of the component model